REMARKS

Reconsideration of the application is requested.

Applicant appreciatively acknowledges the Examiner's acceptance of applicant's drawings previously filed on August 23, 2004. Moreover, applicant further acknowledges the Examiner's withdrawal of previous objections relating to the drawings, the oath/declaration, the specification, the claims, and the Examiner's withdrawal of the claim rejections under 35 U.S.C. § 112 due to the previously filed amendment.

Claims 1-38 are in the application. Claim 19 has been amended.

Claim Rejections

In "Claim Rejections – 35 USC § 102" item 7 on page 2 of the above-identified final Office Action, claims 9-14, 19, 28-33, and 38 have been rejected as being fully anticipated by U.S. Publication No. 2002/0013779 to *Sridhar* (hereinafter **SRIDHAR**) under 35 U.S.C. § 102(e).

In "Claim Rejections – 35 USC § 103" item 8 on page 6 of the above-identified final Office Action, claims 1-8, 15-18, 20-27, 34-38 have been rejected as being obvious over **SRIDHAR** in view of U.S. Patent No. 5,619,688 to *Bosworth, et al.* (hereinafter **BOSWORTH**) under 35 U.S.C. § 103(a).

As will be explained below, it is believed that the claims were patentable over the cited art in their original form and, therefore, the claims have not been amended to overcome the references.

Before discussing the prior art in detail, it is believed that a brief review of the invention as claimed, would be helpful. Claim 1 calls for a novel method for identifying looked-up table fields in a data processing statement and generating SQL statements to join the target tables to the basis table. More specifically, the novel method of claim 1 includes, *inter alia*, the operations of:

> parsing a data processing statement;

identifying table field or fields referenced in said data processing statement;

for each identified table field, determining whether the table field is a looked-up field;

identifying a basis table of which non-looked up ones of said identified table field or fields are members;

identifying one or more target tables from which said looked-up one or ones of said identified table field or fields are to be looked up;

generating a SQL statement, including with said generated SQL statement field or fields to be selected from said basis table and a FROM clause enumerating said basis table, and if the data processing statement was determined to contain one or more fields to be looked up from one or more target tables, further including among said field or fields to be selected said one or more fields to be looked up from said one or more target tables, and one or more JOIN clauses respectively joining said basis table and said one or more target tables, and one or more corresponding ON clauses respectively specifying one or more corresponding conditions on which rows of said basis and said one or more target tables are to be joined, each of said one or more conditions comprising a corresponding look-up field.

Independent claim 20 contains similar language as claim 1.

Thus, claim 1 and 20 are directed to methods of identifying table fields from a ***parsed data processing statement***, determining if any of the fields are ***looked-up fields***, identifying the corresponding basis table or target tables for each field, and for each looked-up field, generating a ***SQL statement***. In particular, the novel SQL statement generation routine includes a FROM clause to enumerate the basis table containing the look-up field, a JOIN clause to join the basis table to the target table, and an ON clause to specify one of more corresponding conditions regarding which rows of each table are to be joined, where each condition comprises a corresponding look-up field.

Independent claim 9 is drawn to a method of data processing techniques associated with data processing operations involving multiple tables of a relational database. The novel method of claim 9includes, *inter alia*:

> presenting a first plurality of fields of a first table for selection for use in a data processing operation;
> receiving a selection of a first field that is a member of said first fields;
> determining whether said selected first field is a first designated look-up field for looking up first one or more of a second plurality of fields of a second table;
> presenting said second plurality of fields for selection for use in said data processing operation, if it is determined that said selected first field is a first designated look-up field for looking up first one or more of said second plurality of fields of said second table.

Independent claim 28 is directed towards an apparatus, but contains similar language as claim 9.

Thus, claim 9 discloses a method of multi-part data table manipulation. In particular, claim 9 discloses allowing **fields** of a data table to be selected for a data processing operation, **determining** upon a field selection whether the selected field is a **designated look-up field** for looking up a **second group of fields** in another data table, and allowing this second group of fields from the other data table to be selected and used in the data processing operation.

Rejections Under 35 U.S.C. §102(e)

The **SRIDHAR** reference discloses a method of displaying data schema table information and relationships, allowing a web developer to select these displays, and generating the necessary backend logic for the developer to use the selected displays. *See* **SRIDHAR** page 1, paras. 5-9. In particular, **SRIDHAR** teaches "automatically presenting relationship information between a first table and a second table of a database[, and] includes ascertaining an existence of a first foreign key relationship between the first table and the second table." **SRIDHAR**, page 1, para. 9. This presentation of information, however, does not teach the specific method disclosed in claim 9 of the present invention of "presenting ... a first table" and then "determining" if a particularly selected field is a "designated look-up field" for other fields in a separate table.

More specifically, **SRIDHAR**, teaches using one or more "link tables" (See paras. 34. 86-88, and Figs. 1, 9) to describe the relationship between separate tables. The present invention, on the other hand, utilizes the characteristics of a selected field to determine whether the information from a second table needs to be looked-up and presented.

Clearly, **SRIDHAR** does not show "determining whether said selected first field is a first designated look-up field" as recited in claim 9 or claim 28 of the instant application. Because **SRIDHAR** fails to teach this look-up characteristic method, **SRIDHAR** cannot anticipate claim 9 or claim 28.

Rejections Under 35 U.S.C. §103(a)

In contrast to the present invention, **SRIDHAR** teaches "a computer-implemented method for facilitating website development by a website developer from a supplied data schema." **SRIDHAR** page 1, para. 5. More specifically, **SRIDHAR** teaches methods of allowing the website developer to select between various data views generated from the data schemas, generating backend logic to support each data view, alternately generating the data views from a user data model, allowing the user to select presented relationships between tables in the data schema, and automatically extracting the SQL statements from the selected relationships. *See* **SRIDHAR** page 1, paras. 5-9.

In contrast, claim 1 specifies that the "table field or fields" are identified in a ***parsed data processing statement***. The final Office Action asserts that **SRIDHAR** discloses parsing a data statement in para. 29, Figs. 3-4 and 14. Applicant respectfully traverses.

Applicant asserts that the cited references do not teach using a ***parsed data processing statement*** to identify table fields. In particular, the extracted data that is manipulated in paragraph 29 of **SRIDHAR** is merely data selected by the user from a plurality of data views to be extracted for a separate data output. As these ***views*** in **SRIDHAR** are either generated from "all possible user ***data models***" that are generated from the data schema or from edited developer-specified ***data models***, which can not read on "a data processing ***statement***" that specifies one or more table fields as recited in claim 1 of the instant application. In addition, because **SRIDHAR** infers all links between tables in generating the user data models (**SRIDHAR**, page 2, para. 29 – page 3, para. 30), **SRIDHAR** can not teach "determining whether the table field is a look up table" as recited in claim 1 of the present application. More specifically, with these pre-generated or edited data models of **SRIDHAR**, a determination of whether a particular field is a look-up field is useless, because all links have already been inferred/created in generating all data models or already specified by the developer in using a specifically edited model.

In addition, the final Office Action states that the supplier and parts table of Figure 4 of **SRIDHAR** reads on the basis and target tables specified by claim 1, respectively. However, **SRIDHAR** teaches that these two tables are linked by a link table (Fig. 1) which describes the attributes of the relationship between supplier and parts. *See* **SRIDHAR** page 3, para. 34. Thus, unlike the present invention, which determines whether a specified table field is a designated look-up field that references another set of fields in a second table, **SRIDHAR** teaches using a **third** link table between the two data tables to store information about relationships between tables. *See* e.g. **SRIDHAR** Figure 1, page 7, para. 65.

Moreover, the ***identification*** of a link table or "join terms" as indicated in **SRIDHAR** can not read on the ***generation*** of a SQL statement with JOIN clauses as recited in claim 1 of the instant application, because they are separate functions.

Additionally, Examiner acknowledges that Sridhar does not teach or otherwise disclose including with a ***generated SQL statement***: field or fields to be selected from a basis table and a FROM clause enumerating said basis table, and one or more corresponding ON clauses respectively specifying one or more corresponding conditions on which rows of said basis and said one or more target tables are to be joined, each of said one or more conditions comprising a ***corresponding look-up field***. The final office action, however, cites **BOSWORTH** as showing these elements of claim 1 that are deficient in **SRIDHAR**.

In contrast to the present invention, the **BOSWORTH** reference discloses a method of constructing data queries that make changes to data stored in a database in a simple and efficient manner. *See* **BOSWORTH** col. 1, lines 6-10, and col. 2 lines 55-56. In particular, **BOSWORTH** does not teach or otherwise disclose determining whether a selected table field is a looked-up field that references other fields in a separate table, nor does it teach automatically generating SQL statements. As such, it does not cure the above mentioned deficiencies of **SRIDHAR**.

Even assuming, *arguendo*, that **SRIDHAR** did not contain the above mentioned deficiencies, **BOSWORTH** does not teach the subject matter of the acknowledged deficiencies in **SRIDHAR**. In particular, **BOSWORTH**'s use of the SQL directives, as cited in the office action, relate to the formation of a query table from two tables, and does not teach nor provide a method for generating SQL statements to present looked-up fields from a target table to be selected along with fields from a basis table. In particular, the fact that **BOSWORTH** shows a set of SQL expressions in demonstrating a query with references to two tables, does not teach or anticipate the specific method of the claimed invention, because the novel method of claim 1 specifies an arrangement of SQL statements to present looked-up fields from a target table to be selected along with fields from a basis table.

For example, an instructional book on how to use C++ directives would not read on a novel computer program even though all the terms of the program are taught in the reference, because the novel method of the program is the arrangement of the terms in a manner to achieve a desired result. Thus, since **BOSWORTH** merely discusses how some SQL terms would work in a query, he cannot teach the separate arrangement of those terms in claim 1. Thus, for at least the reasons set out above, and the deficiencies of **SRIDHAR**, claim 1 is non-obvious and patentable over **SRIDHAR**. **BOSWORTH** does not cure the above discussed deficiencies nor the acknowledged deficiencies of **SRIDHAR**, therefore claim 1 remains patentable over **SRIDHAR**, even when combined with **BOSWORTH**, and is in proper form for allowance.

Clearly, the combination of **SRIDHAR** and **BOSWORTH** does not show "determining whether the table field is a looked-up field" as recited in claim 1 of the instant application.

It is accordingly believed to be clear that none of the references, whether taken alone or in any combination, either show or suggest the features of claims 1, 9, 20, or 28. Claims 1, 9, 20, and 28 are, therefore, believed to be patentable over the art. The dependent claims are believed to be patentable as well because they all are ultimately dependent on claims 1, 9, 20, or 28.

In view of the foregoing, reconsideration and allowance of claims 1-38 are solicited.

In the event the Examiner should still find any of the remaining claims to be unpatentable, counsel would appreciate receiving a telephone call so that, if possible, patentable language can be worked out. In the alternative, the entry of the amendment is requested, as it is believed to place the application in better condition for appeal, without requiring extension of the field of search.

If an extension of time is required, petition for extension is herewith made. Any extension fee associated therewith should be charged to the Deposit Account of Schwabe, Williamson and Wyatt, P.C., No. 50-0393.

Please charge any other fees that might be due with respect to Sections 1.16 and 1.17 to the Deposit Account of Schwabe, Williamson and Wyatt, P.C., No. 50-0393.

Respectfully submitted,

SCHWABE, WILLIAMSON & WYATT, P.C.

Kyle H. Flindt
Reg. No. 42,539

For Applicant

KHF:cgm
March 8, 2005

Schwabe, Williamson & Wyatt, P.C.
Pacwest Center, Suites 1600-1900
1211 SW Fifth Avenue
Portland, Oregon 97204
Tel.: (206) 622-1711
Fax: (206) 292-0460